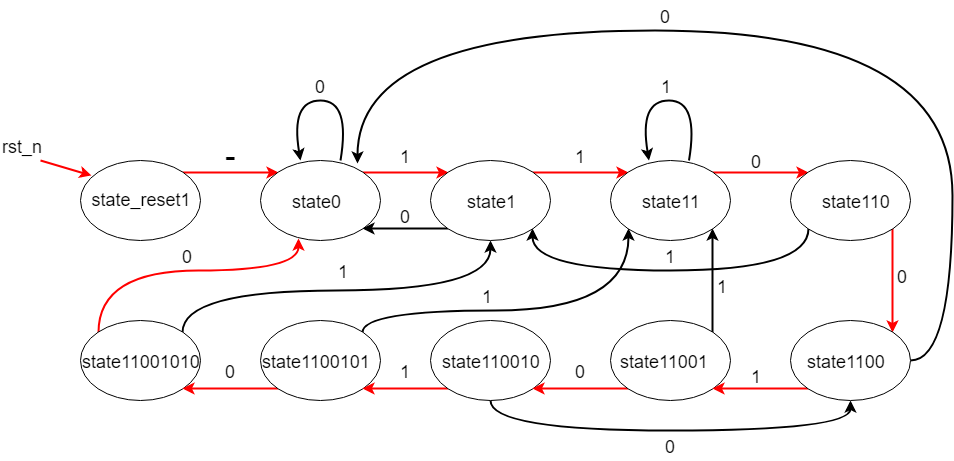
Logic Design

210510210 詹其侁

Lab3

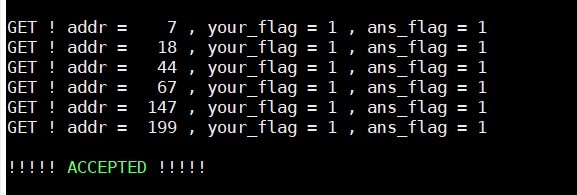
1.state的設計&解釋



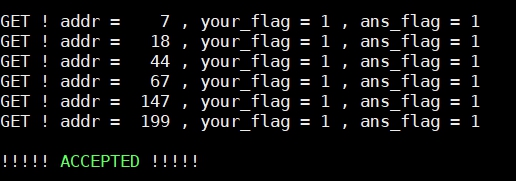
* State\_reset1 : 一開始的state，當rst\_n等於0的時候，會進到裡面。此時傳進來addr=1023的data，這裡會將data設成next\_length，並且將next\_length經由DFF存在length。Next\_addr1會被重置成0，並且無條件進入state0，state\_reset1的作用就是讀取length並且給他一個addr。這裡的flag跟fin都無條件是0，因為他只是作為一個初始化的state。
* State0 : 當從state\_reset1跑到這裡後，就會開始讀取資料，當讀到的資料是0就會不停循環，當讀到的資料是1就會傳給下一個state，此時還沒有讀出完整我們要的信號，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State1 : 這個state紀錄的是”1”，當他讀到1時，會傳給state11，讀到0時，因為我們要的數列是，所以會重新回到state0，此時信號依然不完整，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State11 : 這個state紀錄的是”11”，讀到1時，就會在裡面循環，讀到0時會傳給下一個state，state110，此時信號依然不完整，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State110 : 這個state紀錄的是”110”，讀到0時，會進到下個state，state1100，讀到1時，1101會變成1，所以要回到state1。此時信號依然不完整，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State1100 : 這個state紀錄的是”1100”，讀到1時，會進到下個state，state11001，讀到0時，會跳回去state0重新來過。此時信號依然不完整，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State11001 : 這個state紀錄的是”11001”，讀到0時，會進到下個state，state110010，讀到1時，110011會變成11所以會跳回state11。此時信號依然不完整，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State110010 : 這個state紀錄的是”110010”，讀到1時，會進到下個state，state1100101，讀到0時，1100100會變成0所以會跳回state0。此時信號依然不完整，所以flag是0，假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State1100101 : 這個state紀錄的是”1100101”，讀到0時，會進到下個state，state11001010，同時由於信號已經完整了，所以就會將flag拉起變成1。讀到1時，11001011會變成11所以會跳回state11。假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。
* State11001010 : 這個state紀錄””，進到這個state就代表已經找到我們要的信號，並且flag已經舉過手了，所以flag會是0，同時會再讀新的data，讀到0時，會回到state0，讀到1時，會回到state1。假如addr > length那fin就要拉起來變成1，不然就是0。Addr要加1，同時我們也需要紀錄長度，next\_length = length。

2. Ncverilog Simulation Result:

• Before Design Vision synthesis:



• After Design Vision synthesis:

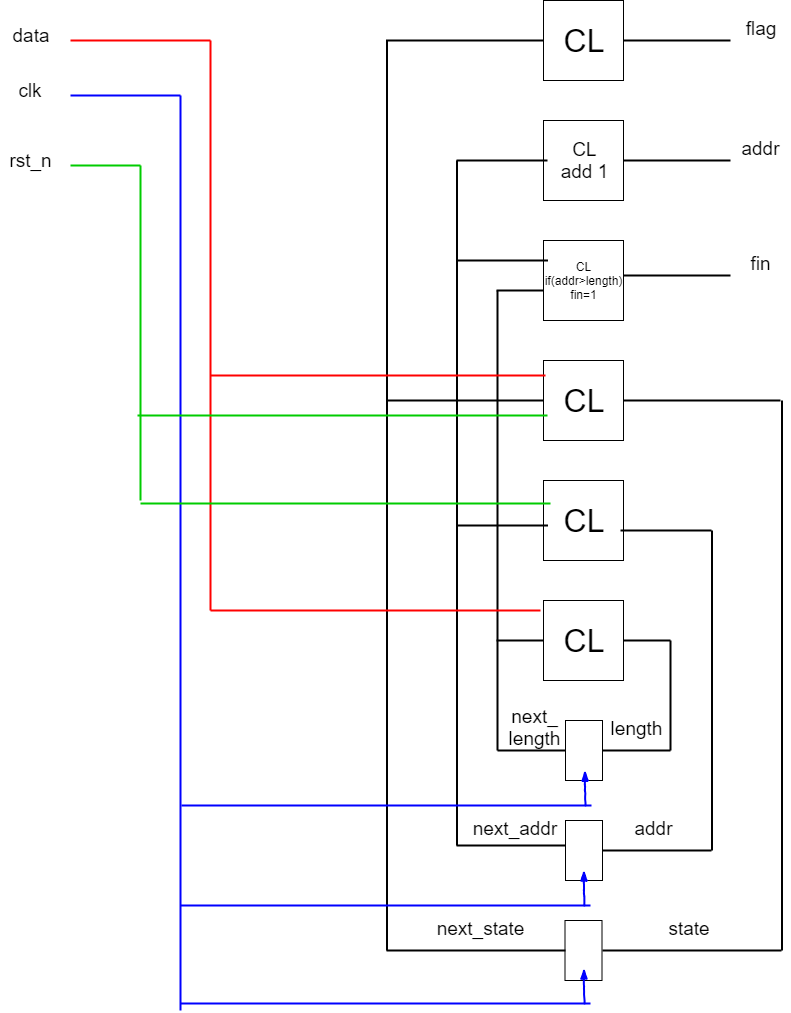


3. nWave Result:



* 紅色圈圈是flag拉起來的地方，rst\_n在第二個cycle變成0其他都是1，其他部分都沒什麼問題。

4. Block Diagram:



這是一個moore machine的block diagram，因為我的input是進到state去做判斷，再把結果給output。這個圖用到三個DFF分別去記，length, addr, state，這三個output判斷的方法，flag是看他在哪個state，fin是看addr是不是大於length，addr則是要加一。

5. Problem encountered and discussion:

* 問題 : 這次我一開始就卡在怎麼從記憶體裡面先找到1023這個位置然後拿出data，一開始我沒想到要用一個state\_reset1所以一直想不通，如果在某個state設值，那length不是就每次都會被重置嗎

解決辦法 : 後來去問助教，想到可以利用state的特性來處理，又加了一個state。

* 問題 : 再工作站跑程式的時候，陷入到無窮迴圈。

解決辦法 : 一開始我跟助教都以為是fin沒拉好的問題，中間也改了很多次，有時候沒有無窮迴圈了，有時候還是一直卡住。改到大家都走了，才發現，是addr的問題，所以後來我再reset那邊加了一個給addr值的assign就成功解決了這個問題。

* 問題 : 我的結果跟答案差了一個cycle。

解決辦法 : 本來我都是在最後一個state把flag拉成1，可是我發現，他在倒數第二個state就應該要拉了，因為他接收到0，就可以確定已經對了，沒必要再跑到最後一個state再拉。搞懂這個概念後，就能理解為什麼會差一個cycle了。

* 問題 : [設定 clock period 的問題](javascript:getPost(225471,%201))，我在跑make syn的時候出現了 ”too many module instance parameter assignment” 的錯誤。

解決辦法 : 我發現是在燒電路的時候，燒成了DFF的電路，所以會錯。在同學的幫助下，我燒了PD的電路就對了。

* 討論 : 這次的lab感覺code沒有上次的多，但是要理解的東西沒有比較少，在做的過程中，我一開始根本不知道DFF在做什麼，看了教學也不太清楚，後來在不停的討論中，了解他可以拿來記錄這次的結果，拿到下次用。這次要用到nWave來debug，很好的幫助我在一次複習老師上課都繪畫的圖形，這次真的蠻謝謝助教跟同學的，如果沒有討論可能會做不出來，因為下禮拜還有考試，所以我這次就沒有做bonus，有點可惜。再次感謝助教跟同學的幫助。